

## CHAPTER

## 36

## SAS I/O Engines

---

<i>Definition</i>	511
<i>Specifying a Different Engine</i>	511
<i>How Engines Work with SAS Files</i>	511
<i>Engine Characteristics</i>	513
<i>Read/Write Activity</i>	513
<i>Access Patterns</i>	514
<i>Levels of Locking</i>	514
<i>Asynchronous I/O or Task Switching</i>	515
<i>Indexing</i>	515
<i>Library Engines</i>	515
<i>Definition</i>	515
<i>Native Library Engines</i>	515
<i>Interface Library Engines</i>	516
<i>Interface View Engines</i>	517

---

### Definition

#### *engines*

are sets of internal instructions that SAS uses to read from and write to files. Engines open files, direct input/output operations, and gather descriptive information about files and their contents. Multiple engines can supply data to and receive data from DATA steps or procedures.

---

### Specifying a Different Engine

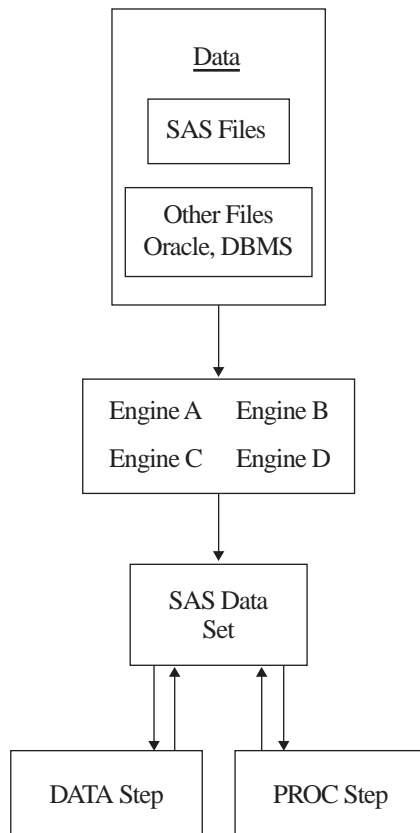
Usually you do not have to specify an engine. SAS automatically selects the appropriate engine. However, you can override the default by specifying an engine name in a LIBNAME statement or by using the ENGINE= system option.

*Operating Environment Information:* The rules for specifying native library engines can vary with the operating environment. Refer to the SAS documentation for your operating environment for details. △

---

### How Engines Work with SAS Files

Figure 36.1 on page 512 shows how SAS data sets are accessed through an engine.

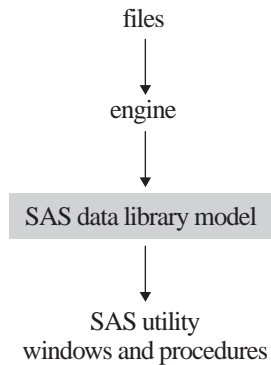
**Figure 36.1** How SAS Data Sets Are Accessed

- Your data is stored in files for which SAS provides an engine. When you specify a SAS data set name, the engine locates the appropriate file or files.
- The engine opens the file and obtains the descriptive information that is required by SAS, for example, which variables are available and what attributes they have, whether the file has special processing characteristics such as indexes or compressed observations, and whether other engines are required for processing. The engine uses this information to organize the data in the standard logical form for SAS processing.
- This standard form is called the *SAS data file*, which consists of the descriptor information and the data values organized into columns (referred to as “variables”) and rows (referred to as “observations”).
- SAS procedures and DATA step statements access and process the data only in its logical form. During processing, the engine executes whatever instructions are necessary to open and close physical files and to read and write data in appropriate formats.

Just as data that is accessed by an engine is organized into the SAS data set model, groups of files that are accessed by an engine are organized in the correct logical form for SAS processing. Once files are accessed as a SAS data library, you can use SAS utility windows and procedures to list their contents and to manage them. See Chapter

26, “SAS Data Libraries,” on page 385 for more information about SAS data libraries. Figure 36.2 on page 513 shows the relationship of engines to SAS data libraries.

**Figure 36.2** Relationship of Engines to SAS Data Libraries



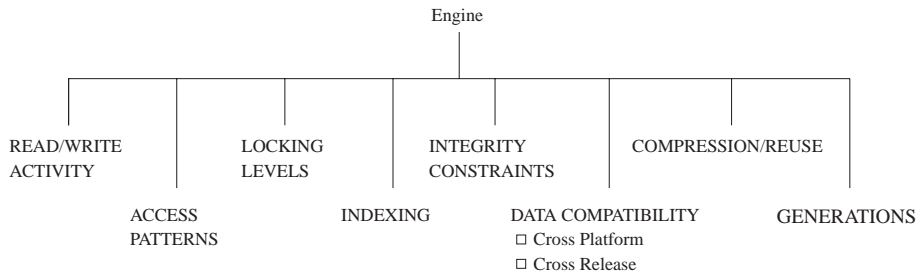

---

## Engine Characteristics

The engine that is used to access a SAS data set determines its processing characteristics. Different statements and procedures require different processing characteristics. For example, the FSEDIT procedure requires the ability to update selected data values, and the POINT= option in the SET statement requires random access to observations and the ability to calculate observation numbers from record identifiers within the file.

Figure 36.3 on page 513 describes the types of activities that engines regulate.

**Figure 36.3** Activities That Engines Regulate




---

## Read/Write Activity

An engine can

- limit read/write activity for a SAS data set to read-only

- fully support updating, deleting, renaming, or redefining the attributes of the data set and its variables
- support only some of these functions.

For example, the engines that access BMDP, OSIRIS, or SPSS files support read-only processing. Some engines that access SAS data views permit SAS procedures to modify existing observations while others do not.

---

## Access Patterns

SAS procedures and statements can read observations in SAS data sets in one of four general patterns:

sequential access	processes observations one after the other, starting at the beginning of the file and continuing in sequence to the end of the file.
random access	processes observations according to the value of some indicator variable without processing previous observations.
BY-group access	groups and processes observations in order of the values of the variables specified in a BY statement.
multiple-pass	performs two or more passes on data when required by SAS statements or procedures.

If a SAS statement or procedure tries to access a SAS data set whose engine does not support the required access pattern, SAS prints an appropriate error message in the SAS log.

---

## Levels of Locking

Some features of SAS require that SAS data sets support different levels at which update access is allowed. When a SAS data set can be opened concurrently by more than one SAS session or by more than one statement or procedure within a single session, the level of locking determines how many sessions, procedures, or statements can read and write to the file at the same time. For example, with the FSEDIT procedure, you can request two windows on the same SAS data set in one session. Some engines support this capability; others do not.

The levels supported are record level and member (data set) level. Member-level locking allows read access to many sessions, statements, or procedures, but restricts all other access to the SAS data set when a session, statement, or procedure acquires update access. Record-level locking allows concurrent read access and update access to the SAS data set by more than one session, statement, or procedure, but prevents concurrent update access to the same observation. Not all engines support both levels.

By default, SAS provides the greatest possible level of concurrent access possible, while guaranteeing the integrity of the data. In some cases, you might want to guarantee the integrity of your data by controlling the levels of update access yourself. Use the CNTLLEV= data set option to control levels of locking. CNTLLEV= allows locking at three levels:

- library
- data set
- observation.

Here are some situations in which you should consider using CNTLLEV=:

- your application controls access to the data, such as in SAS Component Language (SCL), SAS/IML software, or DATA step programming

- you access data through an interface engine that does not provide member-level control of the data.

For more information on the CNTLLEV= data set option, refer to *SAS Language Reference: Dictionary*.

*Note:* SAS software products, such as SAS/ACCESS and SAS/SHARE, contain engines that support enhanced session management services and file locking capabilities. △

---

## Asynchronous I/O or Task Switching

The base SAS software engine and other engines are able to process several different tasks concurrently. For example, you may be entering statements into the Program Editor at the same time that PROC SORT is processing a large file. The reason that this is possible is that the engine allows task switching. Task switching is possible because the engine architecture supports the ability to start one task before another task is finished, or to handle work “asynchronously”. This ability allows for greater efficiencies during processing and often results in faster processing time. Two system options, SYNCHIO and ASYNCHIO control this activity. For more information see the *SAS Language Reference: Dictionary*.

---

## Indexing

One major processing feature of the SAS data model is the ability to access observations by the values of key variables with indexes. See “SAS Indexes” on page 433 for more information on using indexes in SAS data sets. Not all engines support indexing.

---

# Library Engines

---

## Definition

*library engines*

support the SAS data library model. Library engines can be classified as *native* or *interface*.

---

## Native Library Engines

*Native library engines* are engines that read from or write to files formatted by SAS only. Five native library engines are common to all operating environments:

base engine

writes SAS data libraries to disk format. If you do not specify an engine name on the LIBNAME statement when creating a new SAS data library, SAS

automatically selects this engine. The base engine is also automatically selected if you are accessing existing SAS data sets on disk.

The base engine

- is the only engine that supports the full functionality of the SAS data set and the SAS data library.
- supports view engines.
- meets all the processing characteristics required by SAS statements and procedures.
- creates, maintains, and uses indexes.
- reads and writes compressed (variable-length) observations. SAS data sets created by other engines have fixed-length observations.
- assigns a permanent buffer size to data sets and temporarily assigns the number of buffers to be used when processing them.
- repairs damaged SAS data sets, indexes, and catalogs.
- enforces integrity constraints, creates backup files and creates audit trails.

remote engine

allows access to data across SAS session boundaries and across operating environment boundaries.

compatibility engine

enables you to access SAS data sets that were created by older versions of SAS without converting them. SAS determines whether the library is stored in disk or tape format, and automatically reads from and writes to the library in the correct format.

sequential engine

uses a simpler format to access files on storage media that do not allow random access methods, for example, tape or sequential format on disk.

transport engine

enables moving your SAS data sets from one operating environment to another and from one release to another.

*Operating Environment Information:* In some operating environments, one compatibility engine reads both disk and tape. Other operating environments have two separate compatibility engines—one for each storage medium. See the SAS documentation for your operating environment for the engine names and examples for using them.  $\triangle$

---

## Interface Library Engines

*Interface library engines* read from files formatted by other software.

**SPSS**

reads SPSS Release 9 files and SPSS-X files in either compressed or uncompressed format. The engine can also read the SPSS Portable File Format, which is analogous to the transport format for SAS data sets.

**OSIRIS**

reads OSIRIS data and dictionary files in EBCDIC format.

**BMDP**

reads BMDP save files.

---

## Interface View Engines

*Interface view engines* are supported by SAS/ACCESS software. These engines enable you to read and write data directly to and from files formatted by a database management system (DBMS), such as DB2 and ORACLE. Interface view engines enable you to use SAS procedures and program statements to process data values stored in these files without the cost of converting and storing them in files formatted by SAS. Contact your SAS software representative for a list of the SAS/ACCESS interfaces available at your site. For more information about SAS/ACCESS features, see Chapter 33, “Accessing Data in a DBMS,” on page 487 and the SAS/ACCESS documentation for your DBMS.

*Operating Environment Information:* The capabilities and support of these engines vary depending on your operating environment. See the SAS documentation for your operating environment for more complete information.  $\Delta$





The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS Language Reference: Concepts*, Cary, NC: SAS Institute Inc., 1999. 554 pages.

**SAS Language Reference: Concepts**

Copyright © 1999 SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-441-1

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute, Inc.

**U.S. Government Restricted Rights Notice.** Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, November 1999

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries.® indicates USA registration.

IBM, ACF/VTAM, AIX, APPN, MVS/ESA, OS/2, OS/390, VM/ESA, and VTAM are registered trademarks or trademarks of International Business Machines Corporation. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.